Machine Aid for Switching Circuit Design*

CLAUDE E. SHANNON†, FELLOW, IRE AND EDWARD F. MOORE†, ASSOCIATE, IRE

Summary—**The design of circuits composed of logical elements may be facilitated by auxiliary machines. This paper describes one such machine, made of relays, selector switches, gas diodes, and germanium diodes. This machine (called the relay circuit analyzer) has as inputs both a relay contact circuit and the specifications the circuit is expected to satisfy. The analyzer (1) verifies whether the circuit satisfies the specifications, (2) makes systematic attempts to simplify the circuit by removing redundant contacts, and also (3) obtains mathematically rigorous lower bounds for the numbers and types of contacts needed to satisfy the specifications. A special feature of the analyzer is its ability to take advantage of circuit specifications which are incompletely stated. The auxiliary machine method of doing these and similar operations is compared with the method of coding them on a general-purpose digital computer.**

Introduction

SOME OPERATIONS which assist in the design of relay circuits or other types of switching circuits can be described in very simple form, and machines can be constructed which perform them more quickly and more accurately than a human being can. It seems possible that machines of this type will be useful to those whose work involves the design of such circuits.

The present machine, called the relay circuit analyzer, is intended for use in connection with the design of two terminal circuits made up of contacts on at most four relays. The principles upon which this machine is based are not limited to two terminal networks or to four relays, although an enlarged machine would require more time to operate. Each addition of one relay to the circuits considered would approximately double the size of the machine and quadruple the length of time required for its operation. This type of machine is not applicable to sequential circuits, however, so it will be of use only in connection with parts of the relay circuits which contain contacts, but no relay coils.

Operation of the Machine

The machine, as can be seen from Fig. 1, contains sixteen 3-position switches, which are used to specify the requirements of the circuit. One switch corresponds to each of the $2^4=16$ states in which the four relays can be put. Switch number two in the upper right hand corner, for instance, is labeled $W+X+Y'+Z$, which corresponds to the state of the circuit in which the relays labeled W, X, and Z are operated, and the relay labeled Y is released.

The three positions of this switch correspond to the requirements which can be imposed on the condition of the circuit when the relays are in the corresponding state. Since any single relay contact circuit assumes only one of two values (open or closed) the inclusion of a third value (doesn't matter, don't care, or vacuous, as it has been called by various persons) merits some explanation. If the machine, of which the relay circuit being designed is to be a part, only permits these relays to take on a fraction of the 2^n combinations of which n relays are capable, then it will not matter what the circuit does in the excluded states. The user indicates the optional nature of these requirements by placing the corresponding switches in the "don't care" positions. With these options, there is a wider range of circuits satisfying the requirements, and hence more likelihood that a circuit with fewer contacts will suffice. The sixteen 3-position switches thus permit the user not only to require the circuit under consideration to have exactly some particular hindrance function, but also allow the machine more freedom in the cases where the circuit need not be specified completely.

In order to make a machine of this type to deal with n relays (this particular machine was made for the case $n=4$), 2^n such switches would be required, corresponding to the 2^n states n relays can assume. In each of these states the circuit can be either open or closed, so there are 2^{2^n} functionally distinct circuits. But since each switch has 3 positions, there are 3^{2^n} distinct circuit requirements specifiable on the switches, which in the case $n=4$ amounts to 43,046,721. Thus, the number of problems which the analyzer must deal with is quite large, even in the case of only four relays.

The left half of the front panel of the machine (see Fig. 1) is a plugboard on which the circuit being analyzed can be represented. There are three transfers from each of the four relays, W, X, Y, and Z brought out to jacks on this panel, and jacks representing the terminals of the network are at the top and bottom. Using patch cords, it is possible to plug up any circuit using at most three transfers on each of the four relays. This number of contacts is sufficient to give a circuit representing any switching function of four variables.

If the specifications for the circuit have been put on the sixteen switches, and if the circuit has been put on the plugboard, the relay circuit analyzer is then ready to operate.

With the main control switch and the evaluate-compare switch both in the "evaluate" position, pressing the start button will cause the analyzer to evaluate the circuit plugged in, i.e. to indicate in which of the states the circuit is closed by lighting up the corresponding indicator lamps.

Turning the evaluate-compare switch to the "compare" position, the analyzer then checks whether the

* Decimal classification: 621.375.2×R257. Original manuscript received by the Institute, May 28, 1953; revised manuscript received June 29, 1953.

† Bell Telephone Laboratories, Inc., Murray Hill, N. J.

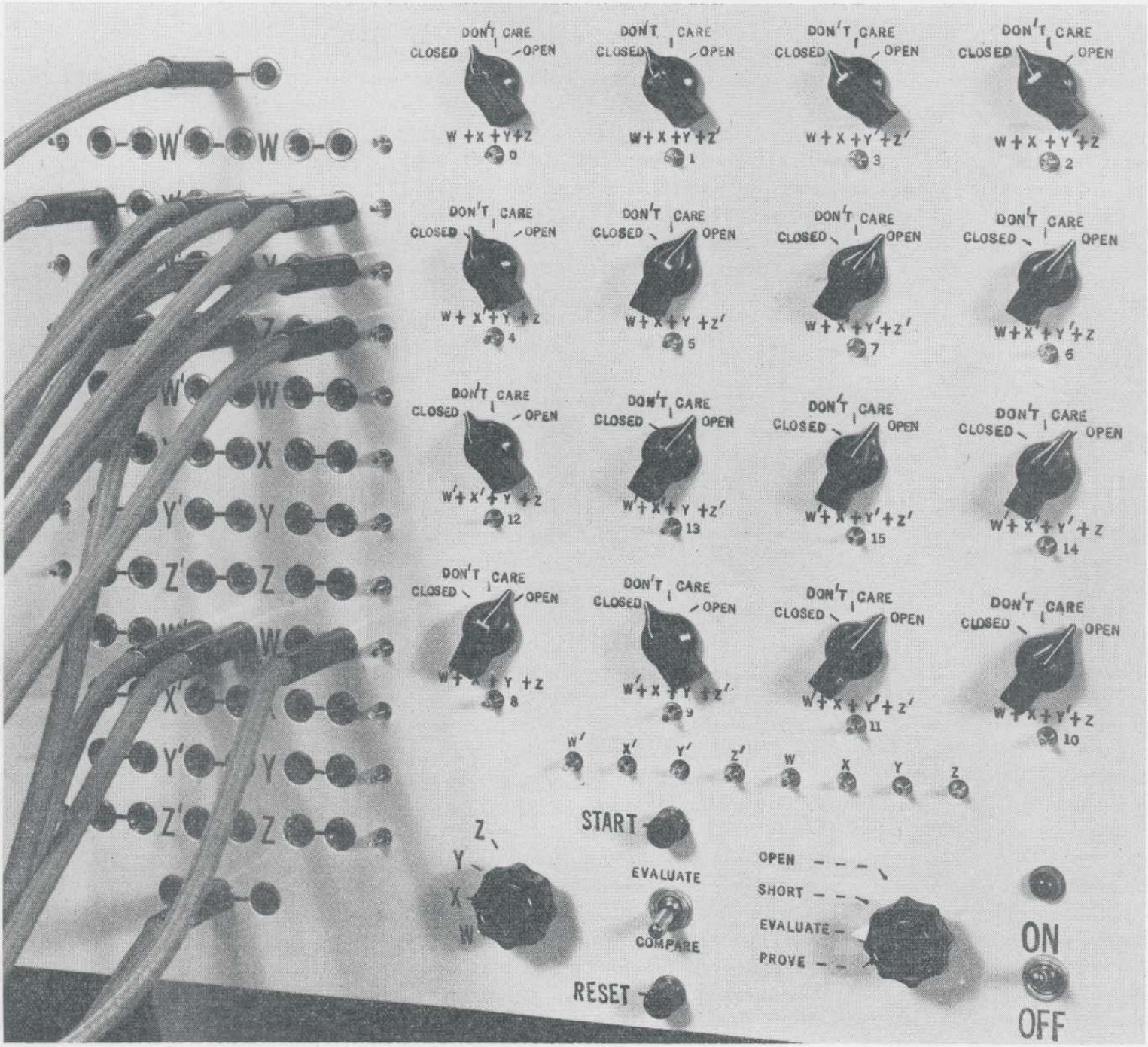

Fig. 1—View of front panel of the relay circuit analyzer.

circuit disagrees with the requirements given on the switches. A disagreement is indicated by lighting the lamp corresponding to the state in question. If a switch is set for closed and the actual circuit is open in that state, or vice versa, a disagreement is indicated, but no disagreement is ever registered when the switch is set in the "don't care" position, regardless of the circuit condition. The compare position, while it gives information equivalent to that given in the evaluate position, gives it in a form more convenient for noticing errors.

After a circuit has been found which agrees entirely with the requirements, the main control switch is then turned to the "short test" position and the start button is pressed again. The machine then determines whether any of the contacts in this circuit could have been shorted out, with the circuit still satisfying the requirements. The machine indicates on the lamps beside the contacts which ones have this property.

It may be surprising to the reader that anyone would ever need the assistance of a machine to find a contact which could be shorted out without affecting its circuit. While this is certainly true of simple examples, in more complicated circuits such redundant elements are often far from obvious (particularly if there are some states for which the switches are in the "don't care" position, since the simplified circuit may be functionally different from the original one, as long as it differs only in the "don't care" state). It is often quite difficult to see the simplification in these cases.

The analyzer is also helpful in case the circuit being analyzed is a bridge, because of the complications involved in tracing out all paths in the bridge. The circuit

shown in Fig. 2 is an example of a circuit which was not known to be inefficiently designed until put on the analyzer. It determined in less than two minutes (including the time required to plug the circuit into the plugboard) that one of the contacts shown can be shorted out. How likely would a human being be to solve this same problem in the same length of time?

After the short test has been performed, putting the main control switch in the "open test" position permits the analyzer to perform another analogous test, this time opening the contacts one at a time.

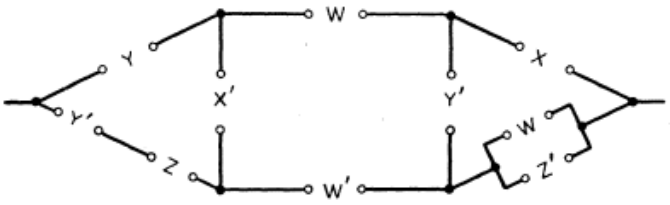

Fig. 2—The relay circuit analyzer was able to simplify this circuit, removing one contact, in less than two minutes total time. Can you do as well?

These two particular types of circuit changes were chosen because they are easy to carry out, and whenever successful, either one reduces the number of contacts required. There are other types of circuit simplification which it might be desirable to have a machine perform, including various rearrangements of the circuit. These would have required more time as well as more equipment to perform, but would probably have caused the machine to be more frequently successful in simplifying the circuit. Using such techniques, it might be possible to build a machine which could design circuits efficiently starting from basic principles, perhaps by starting with a complete Boolean expansion for the desired function and simplifying it step by step. Such a machine would be rather slow (unless it were built to operate at electronic speeds, and perhaps even in this case), and not enough planning has been done to know whether such a machine is practically feasible. However, the fact that such a machine is theoretically possible is certainly of interest, whether anyone builds one or not.

Another question of theoretical interest is whether a logical machine could be built which could design an improved version of itself, or perhaps build some machine whose over-all purpose was more complicated than its own. There seems to be no logical contradiction involved in such a machine, although it will require great advances in the general theory of automata before any such project could be confidently undertaken.

To return to the relay circuit analyzer, a final operation which it performs is done with the main control switch in the *prove* position. Pressing the start button and moving the other 4-position switch successively through the W, X, Y and Z positions, then certain of the eight lamps W, W', X, X', Y, Y', Z, Z' will light up. The analyzer has carried out a proof as to which kinds of contacts are required to synthesize the function using the method of reduction to functions of one variable, which will be explained in a forthcoming paper. The analyzer here ignores whatever circuit has been plugged in the plugboard, and considers only the function specified by the sixteen 3-position switches. If every circuit which satisfied these specifications requires a back contact on the W relay, the W' light will go on, etc.

If, for instance, seven of the eight lights are on, any circuit for the function requires at least seven contacts, and if there is, in fact, a circuit which uses just seven, the machine has in effect, given a complete proof that this circuit is minimal. Circuits for which the machine can give such a complete proof are fairly common, although there are also circuits (which can be shown to be minimal by more subtle methods of proof) which this machine could not prove minimal. An example is the circuit of Fig. 2. This can be simplified by the analyzer to a circuit of nine contacts, but in the *prove* position the analyzer merely indicates that at least eight contacts are necessary. It can be shown by other methods that the 9-contact circuit is minimal. But at any rate, the analyzer always gives a mathematically rigorous lower bound for the number of contacts.

The small size and portability of this machine depends on the fact that a mixture of relay and electronic circuit elements were used. The gas diodes are particularly suited for use where a small memory element having an associated visual display is required, and the relays and selector switches are particularly suited for use where the ability to sequence and interconnect, using only a small weight and space, is required. In all, the relay circuit analyzer uses only 24 relays, 2 selector switches, 48 miniature gas diodes, and 14 germanium diodes as its logical elements.

It may be of interest to those familiar with general purpose digital computers to compare this method of solution of this problem on such a small, special-purpose machine with the more conventional method of coding it for solution on a high-speed general-purpose computer. One basic way in which the two methods differ is in the directness with which the circuits being analyzed are represented. On a general-purpose computer it would be necessary to have a symbolic description of the circuit, probably in the form of a numerical code describing the interconnections of the circuit diagram, and representing the types of contacts that occur in the various parts of the circuit by means of a list of numbers in successive memory locations of the computer. On the other hand, the relay circuit analyzer represents the circuit in a more direct and natural manner, by actually having a copy of it plugged up on the front panel.

This difference in the directness of representation has two effects. First, it would be somewhat harder to use the general-purpose computer, because the steps of translating the circuit diagram into the coded description and of typing it onto the input medium of the

computer would be more complicated and lengthy than the step of plugging up a circuit directly. The second effect is in the relative number of logical operations (and hence, indirectly, the time) required by the two kinds of machines. To carry out the fundamental step in this procedure of determining whether the given circuit (or some modification of it obtained by opening or shorting a contact) is open or closed for some particular state of the relays, requires only a single relay operate time for the relay circuit analyzer. However, the carrying out of this fundamental step on a general-purpose digital computer would require going through several kinds of subroutines many times. There would be several ways of coding the problem, but in a typical one of them the computer would first go through a subroutine to determine whether a given contact were open or closed, repeating this one for each contact in the circuit, and then would go through another subroutine once for each node of the network. Altogether this would probably involve the execution of several hundred orders on the computer, although by sufficiently ingenious coding this might be cut down to perhaps 100. Since each order of a computer takes perhaps 100 times the duration of a single logical operation (i.e., a pulse time, if the computer is clock-driven), actually what takes 1 operation time on one machine takes perhaps 10,000 on another.

Since 10,000 is approximately the ratio between the speed of a relay and of a vacuum tube in performing logical operations, this gain of about 10,000 from the directness of the representation permits this relay machine to be as fast at this kind of problem as a general-pupose electronic computer.

This great disparity between the speeds of a general-purpose and of a special-purpose computer is not typical of all kinds of problems, since a typical problem in numerical analysis might only permit of a speed-up by a factor of 10 on a special-purpose machine (since multiplications and divisions required in the problem use up perhaps a tenth of the time of the problem). However, it seems to be typical of combinatorial problems that a tremendous gain in speed is possible by the use of special rather than general-purpose digital computers. This means that the general-purpose machines are not really general in purpose, but are specialized in such a direction as to favor problems in analysis. It is certainly true that the so-called general purpose machines are logically capable of solving such combinatorial problems, but their efficiency in such use is definitely very low. The problems involved in the design of a general-purpose machine suitable for a wide variety of combinatorial problems seem to be quite difficult, although certainly of great theoretical interest.

Conclusions

An interesting feature of the relay circuit analyzer is its ability to deal directly with logical circuits in terms of 3-valued logic. There would be considerable interest in techniques permitting easy manipulation on paper with such a logic, because of its direct application to the design of economical switching circuits. Even though such techniques have not yet been developed, machines such as this can be of value in connection with 3-valued problems.

Whether or not this particular kind of machine ever proves to be useful in the design of practical relay circuits, the possibility of making machines which can assist in logical design procedure promises to be of value to everyone associated with the design of switching circuits. Just as the slide rule and present-day types of digital computers can help perform part of the routine work associated with the design of linear electrical networks, machines such as this may someday lighten much of the routine work associated with the design of logical circuits.